Semi-Definite Programming as a Simple Extension to Linear Programming: Convex Optimization with Queueing, Equity and Other Telecom Functionals

Jérôme Galtier

France Télécom R&D and Mascotte team (INRIA/CNRS/UNSA)
Sophia Antipolis

We present Semi-Definite Programming (SDP) as an extension of linear programming and the basics of the duality theory. We show that SDP allows to express naturally a large set of flow problems in the telecommunication context with specific notions of fairness.

Keywords: Semi-definite programming, linear matrix inequalities, flow, fairness, equity.

1 Introduction

In the recent years, the telecommunication field has experienced an explosion of services, with as various leading applications as mobility and the Internet. This revolution has impacted both the nature and the volume of the traffic in nowadays networks.

Meanwhile, in an attempt to simplify the management of such networks, the protocols that have been proposed recently aim at unifying the way the traffic is carried. In a sense, both ATM and the IP have tried to be the universal model for transiting all kinds of information.

Logically, both of these protocols have integrated mechanisms to optimize the way the traffic is routed. However, the optimization software is still today something to improve. The issue is becoming more challenging as many users can interfere in a network, and the impact of these on the final quality of service, including the delay and the equity among users for instance, is becoming increasingly important as the load is getting bursty.

In this paper, we introduce the reader to semi-definite programming, and show how complex and new network functionals can be very naturally implemented using this method. After recalling the basics of underlying mathematics and links to linear programming in Section 2, we show in Section 3 how different network functionals can be described as a SDP program. Finally, in Section 4 we mention other significant improvements of the field of algorithms for telecommunication-oriented problems using SDP.

2 Mathematical Background

2.1 Quadratic Forms

Definition 1 *A function φ that maps $\mathbb{R}^n \times \mathbb{R}^n$ to $\mathbb{R}$ is said to be bilinear if*

- $\forall x,y,z \in \mathbb{R}^n \quad \forall \lambda \in \mathbb{R} \qquad \varphi(x+\lambda y,z) = \varphi(x,z) + \lambda\varphi(y,z)$
- $\forall x,y,z \in \mathbb{R}^n \quad \forall \lambda \in \mathbb{R} \qquad \varphi(x,y+\lambda z) = \varphi(x,y) + \lambda\varphi(x,z)$

Now if we denote by e_i the i^{th} vector of the canonical basis of $\mathbb{R}^n$, and write

$$A := [\varphi(e_i,e_j)]_{1\leq i,j\leq n}$$

then we have the following relation:

$$\forall x,y \in \mathbb{R}^n \qquad \varphi(x,y) = x^t Ay.$$

Definition 2 *A function q that maps $\mathbb{R}^n$ to $\mathbb{R}$ is said to be quadratic is there exists some bilinear function φ from $\mathbb{R}^n \times \mathbb{R}^n$ to $\mathbb{R}$ such that:*

$$\forall x \in \mathbb{R}^n \qquad q(x) = \varphi(x,x).$$

In fact, there is a way to check whether a function is quadratic, given in the following.

Proposition 1 *A function q from $\mathbb{R}^n$ to $\mathbb{R}$ is quadratic if and only if*

- $\forall x \in \mathbb{R}^n \quad \forall \lambda \in \mathbb{R}^n \qquad q(\lambda x) = \lambda^2 x$
- $(x,y) \mapsto \frac{1}{2}[q(x+y) - q(x) - q(y)]$ *is bilinear.*

Obviously, then, a quadratic function can also be written as

$$q(x) = x^t Ax,$$

which shows that if is given by $x = (x_1, \ldots, x_n)^t$, then $q(x)$ is an homogeneous polynomial of degree 2, that can be noted as

$$q(x) = \sum_{i=n}^{i=1} \alpha_i x_i^2 + \sum_{1 \le i < j \le n} \alpha_{i,j} x_i x_j.$$

Then A takes the shape of

$$A = \begin{bmatrix} \alpha_1 & & & & \\ & \ddots & & \frac{1}{2}\alpha_{i,j} & \\ & & \ddots & & \\ & \frac{1}{2}\alpha_{i,j} & & \ddots & \\ & & & & \alpha_n \end{bmatrix}.$$

We need two other definitions:

Definition 3 *We say that a quadratic function from $\mathbb{R}^n$ to $\mathbb{R}$ is positive if*

$$\forall x \in \mathbb{R}^n \qquad q(x) \ge 0,$$

and we note $q \succeq 0$ (and by extension $A \succeq 0$ if A is the symmetric matrix representation of q in some basis).

By extension, We write $q_1 \succeq q_2$ if $q_1 - q_2 \succeq 0$. We also denote U_{psd} the set of positive square matrices $n \times n$ over $\mathbb{R}$.

Definition 4 *We say that a positive quadratic function from $\mathbb{R}^n$ to $\mathbb{R}$ is definite (and we write $q \succ 0$) if*

$$\forall x \in \mathbb{R}^n \qquad q(x) = 0 \Rightarrow x = 0.$$

Proposition 2 *Let q be a positive quadratic function. Then q is definite if and only if its symmetric matrix representation A is non-degenerated.*

PROOF. Obviously if q is definite, then for all $x \neq 0$, we have $x^t Ax > 0$ and therefore $Ax \neq 0$.

Reversely, suppose q definite and A non degenerated. Then if $x \neq 0$, we have $Ax \neq 0$ and for some y, $y^t Ax \neq 0$. Meanwhile $q(x + \lambda y) = x^t Ax + 2\lambda y^t Ax + \lambda^2 y^t Ay \ge 0$ for all $\lambda \in \mathbb{R}$, and therefore the polynomial form in λ hasn't two distinct real roots. In terms of discriminent,

$$(y^t Ax)^2 - q(x)q(y) \le 0,$$

and since $y^t Ax \neq 0$, we have necessarily $q(x) > 0$. □

By abuse of language, we therefore say that a positive quadratic function q is semi-definite (whether it is definite or not). A semi-definite quadratic function is necessarily either positive or negative.

Proposition 3 *Let q be a positive definite quadratic function. Then there exists $\eta > 0$ such that $q(x) \ge \eta x^t x$, for $x \in \mathbb{R}^n$.*

PROOF. The set $\{x \in \mathbb{R}^n, x^t x = 1\}$ is compact, and q is continuous. Therefore $\{q(x) \ : \ x \in \mathbb{R}^n, x^t x = 1\}$ is closed, and its lower bound is positive and non zero (otherwise q is not definite). Let η be the lower bound. □

2.2 Semi-Definite Programming

We put here some basics of SDP. For more details, see [2]. Consider the following couple of semidefinite programs :

$$Sdp = \begin{cases} Max & C\cdot X \\ s.c. & A_i \cdot X = b_i \quad \forall i \in \{1,\dots,m\} \\ & X \in U_{psd} \end{cases} \quad Dsdp = \begin{cases} Min & y^t b \\ s.c. & \sum_{i=1}^{m} y_i A_i - C \succeq 0 \end{cases}$$

Lemma 1 *(**Slater**) If there existe a vector $y \in \mathrm{I\!R}^m$ such that $\sum_{i=1}^{m} y_i A_i - C \succ 0$ then the program Sdp has a bounded optimal value.*

PROOF. Suppose there exists a strictly feasible solution of the dual $Dsdp$: $S = \sum_{i=1}^{m} y_i A_i - C \succ 0$. Let η be the associated constant of propostion 3. Let X be a solution of Sdp such that $C\cdot X \geq v$, then :

$$\eta\ (X \cdot Id) \leq X \cdot S = y^t b - C \cdot X \leq y^t b - v$$

Thus the sum of the diagonal entries of X is bounded then all the solutions of Sdp are bounded. □

The fact that the polar cone of the cone of positive semidefinite matrices, i.e. the set of matrices Y such that the inner product $Y \cdot X = tr(Y^t X) \geq 0$ is the cone of positive semidefinite matrices itself allows us to check in an easy way an extended to semidefinite programming Farkas lemma an a weak duality theorem.

Lemma 2 *(**Farkas**) Considér m matrices A_i and a vector $b \in \mathrm{I\!R}^m$. Let $y \in \mathrm{I\!R}^m$ be a vector such that $\sum_{i=1}^{m} y_i A_i \succ 0$. Then there exists a matrix $X \in U_{psd}$ such that $A_i \cdot X = b_i,\ \forall i \in \{1,\dots,m\}$ if and only if $y^t b \geq 0$ for all vector y satisfying $\sum_{i=1}^{m} y_i A_i \succeq 0$.*

PROOF. Consider the set $T = \{(A_i \cdot X)_{i\in\{1,\dots,m\}}\ /\ X \in U_{psd}\}$. Our result means T is convex and closed. Oviously T is convex. We show T is closed. Let $(X_n)_{n\in\mathrm{I\!N}}$ be a series such that $A_i \cdot X_n \to b_i$ for $i \in \{1,\dots,m\}$. Take $S = \sum_{i=1}^{m} y_i A_i \succ 0$ and the associated η. Then for some $\varepsilon > 0$, there exists some $N \in \mathrm{I\!N}$ such that for $n \geq N$, $\eta\ Id \cdot X_n \geq S \cdot X_n \geq \sum_{i=1}^{m} y_i b_i + \varepsilon$. So the X_n are bounded and since U_{sdp} is closed, at least a sub-series of the (X_n) connverges to some $X \in T$. □

Theorem 1 *(Weak duality) Suppose that both programs Sdp and $Dsdp$ have feasible solution, then :*

$$C \cdot X \leq y^t b$$

PROOF. $C \cdot X \leq \sum_{i=1}^{m} y_i A_i \cdot X \leq y^t b$ □

Theorem 2 *(Strong duality) Let z_{max} and z_{min} be the optimal values of Sdp and $Dsdp$. Suppose there exists a vector $y \in \mathrm{I\!R}^m$ such that $\sum_{i=1}^{m} y_i A_i \succ 0$, then $z_{max} = z_{min}$.*

In fact, the condition $\sum_{i=1}^{m} y_i A_i \succ 0$ can be even avoided, if one allows to introduce a more complex form of dual program, as described in [8].

PROOF. Suppose that $z_{max} < z_{min}$, then the system :

$$\begin{array}{ll} Max & C \cdot X = z_{min} \\ & A_i \cdot X = b_i \quad \forall i \in \{1,\dots,m\} \\ & X \in U_{psd} \end{array}$$

has no solution. Thus by the Farkas lemma extended to semidefinite programming we may say that there exists a vector $(y_0, y) \in \mathrm{I\!R}^{m+1}$ such that :

$$y_0 C + \sum_{i=1}^{m} y_i A_i \succeq 0 \ \text{ and } \ y_0 z_{min} + y^t b < 0 \tag{1}$$

If $y_0 = 0$, then 1 is equivalent to $\sum_{i=1}^{m} y_i A_i \succeq 0$ and $y^t b < 0$. This implies that Sdp has no solution, and leads to a contradiction.
If $y_0 < 0$ we obtain $-C - \frac{1}{y_0} \sum_{i=1}^{m} y_i A_i \succeq 0$ and $z_{min} + \frac{1}{y_0}(y^t b) > 0$, this implies that z_{min} is not an optimal solution of $Dsdp$.
If $y_0 > 0$ then dividing 1 by y_0 we obtain :

$$C + \frac{1}{y_0} \sum_{i=1}^{m} y_i A_i \succeq 0 \text{ and } z_{min} + \frac{1}{y_0}(y^t b) < 0.$$

And therefore, for some $\varepsilon > 0$,

$$C + \frac{1}{y_0} \sum_{i=1}^{m} y_i A_i \succeq 0 \text{ and } z_{min} + \frac{1}{y_0}(y^t b) < -\varepsilon. \tag{2}$$

By hypothesis z_{min} is the optmal value of $Dsdp$ then there exists y^{min} such that :

$$-C + \sum_{i=1}^{m} y_i^{min} A_i \succeq 0 \text{ and } -z_{min} + b^t y^{min} \leq \varepsilon \tag{3}$$

Adding 2 and 3 we get :

$$\sum_{i=1}^{m} (\frac{y_i}{y_0} + y_i^{min}) A_i \succeq 0 \text{ and } (\frac{y}{y_0} + y^{min})^t b < 0$$

these acts and the Farkas lemma, allow us to say that Sdp has no solution. Hypothesis $z_{max} < z_{min}$ leads to a contradiction. Moreover by the weak duality theorem, we have $z_{max} \leq z_{min}$ and we may conclude $z_{max} = z_{min}$. ☐

2.3 Links to Linear Programming

Given a linear program, how can we build a SDP program with the same solution? For sake of simplicity, we will relate here a linear program to the dual form of an integer program. Let $y_1, \ldots, y_m$ be a set of variables, and we aim at minimizing

$$\sum_{i=1}^{i=p} c_i \cdot y_i$$

under the constraint $\forall j \quad b_{j,0} + \sum_{i=1}^{i=p} b_{i,j} y_i \geq 0$. So we take

$$A_i = \begin{pmatrix} b_{1,i} & & 0 \\ & \ddots & \\ 0 & & b_{p,i} \end{pmatrix} \text{ and } C = \begin{pmatrix} -b_{1,0} & & 0 \\ & \ddots & \\ 0 & & -b_{p,0} \end{pmatrix}.$$

Hence the result. Also links to integer programming have become increasingly important. See for instance [1].

3 The Use of Semi-Definite Programming in the Telecommunication Context

The telecommunication context offers a large set of problems with huge needs of optimization. One of the most classical issues is given by flow problems, where a demand of traffic has to be routed through a network having a fixed topology. This network's nodes are represented by a set of vertices V and its links are emphasized by a set of edges E, each of them associated to a capacity $C_e, e \in E$. The demand of traffic can be given by some positive numbers $d_{u,v}$ associated to a source/destination pair (u, v) in the demand set D.

In this paper, we restrict ourselves to the case where the traffic can be splitted (for a given demand between u and v, several routes can be used between u and v, as opposed to monorouting, that allows only one route), and we try to optimize some global parameter over the network, under the assumption that all the demand can be routed through the network, even leaving some non-zero available bandwidth on each link.

Therefore, a solution of the problem can be given by a flow $\phi^e_{u,v}, e \in E, (u,v) \in D$, and surplus margins $\delta_{u,v}, (u,v) \in D$ and $\kappa_e, e \in E$ satisfying the following set of constraints:

$$\left| \begin{array}{lll} \forall (u,v) \in D \quad \forall e \in E & & \phi^e_{u,v} \geq 0 \\ \forall (u,v) \in D & & \delta_{u,v} \geq 0 \\ \forall e \in E & & \kappa_e \geq 0 \\ \forall (u,v) \in D & & \sum_{w \in V s.t. (u,w) \in E} \phi^{(u,w)}_{u,v} \geq \sum_{w \in V s.t. (w,u) \in E} \phi^{(w,u)}_{u,v} + (1+\delta_{u,v}) d_{u,v} \\ \forall (u,v) \in D \quad \forall x \in V - \{u,v\} & & \sum_{w \in V s.t. (x,w) \in E} \phi^{(x,w)}_{u,v} \geq \sum_{w \in V s.t. (w,x) \in E} \phi^{(w,x)}_{u,v} \\ \forall (u,v) \in D & & \sum_{w \in V s.t. (v,w) \in E} \phi^{(v,w)}_{u,v} + (1+\delta_{u,v}) d_{u,v} \geq \sum_{w \in V s.t. (w,v) \in E} \phi^{(w,v)}_{u,v} \\ \forall e \in E & & \sum_{(u,v) \in D} \phi^e_{u,v} \leq (1-\kappa_e) C_e \end{array} \right. \tag{4}$$

The first inequality guarantees that the variables ϕ emphasize real flow. The second and the third ones guarantee that the surpluses are indeed surpluses. The three following ones force them to carry the demand. In fact, these inequalities usually turn tight since the minimization criteria aims at having a network as light as possible. The last one is the capacity constraint. In many of the following models, it can in fact be dropped, since the minimization criteria and the associated variables contain it.

There are two natural ways to optimize the way the network will carry the load (see [6]):

The network-aware optimization consists in trying to keep the best set of available resources in the network in order to anticipate flow variations and/or future demands. It can be formulated as

$$\text{Maximize } \frac{1}{1-\alpha} \sum_{e \in E} \kappa_e^{1-\alpha}, \quad \alpha \geq 0, \alpha \neq 1.$$

The user-aware optimization will affect as many ressources as possible to the users, while saturating the network. The largest circuits are opened to each individual demand, so that it can be fullfilled as easily as possible. Then it is stated as

$$\text{Maximize } \frac{1}{1-\alpha} \sum_{(u,v) \in D} {\delta_{u,v}}^{1-\alpha}, \quad \alpha \geq 0, \alpha \neq 1.$$

The special case $\alpha = 1$, in both cases, corresponds to what is called the *proportional fairness*, and can be formulated respectively as:

$$\text{Maximize } \prod_{e \in E} \kappa_e,$$

and

$$\text{Maximize } \prod_{(u,v) \in D} \delta_{u,v}.$$

We will show in the following that semi-definite programming allows to implement straightforwardly each of these concepts.

3.1 Various Functionalities

Proposition 4 *Let x, y and z be three positive real numbers. Then*

$$\begin{pmatrix} x & z \\ z & y \end{pmatrix} \succeq 0 \text{ if and only if } xy \geq z^2.$$

In particular, if one sets $z = 1$, then the relation $y \geq 1/x$ allow to obtain constraints of the form

$$y \geq \sum_{i=1}^{i=n} \frac{1}{x_i},$$

and then minimizing y is equivalent to minimizing the right member of the equation, which solves the case $\alpha = 2$. Thanks to an idea of Nemirovski[7], we can also integrate the following series of functions in our model.

Proposition 5 *Let x and y be two real positive number. It is possible, using SDP constraints, to bound x and y by the relation*

$$y \leq x^{k/2^p},$$

with $p \in \mathbb{N}$ and $k \in \{0,\dots,2^p-1\}$.

In other words, if $\alpha < 1$ is approximated by some $1 - k/2^p$, then one can generate constraints of the form

$$y \leq \sum_{i=1}^{i=n} {x_i}^{1-\alpha}$$

and maximizing y is equivalent to maximizing the right member, with solves our problem with very good precision for $0 < \alpha < 1$.

PROOF. Now let $a_1,\dots,a_p$ be a series of $0/1$ integers, such that

$$k = \sum_{i=1}^{i=p} a_i 2^{i-1}.$$

We note $y_0 = 1$, and submit $y_1,\dots,y_p$ to the following contraints:

$$\begin{cases} \begin{pmatrix} y_{i-1} & y_i \\ y_i & x \end{pmatrix} \succeq 0 & \text{if } a_i = 1 \\ \begin{pmatrix} y_{i-1} & y_i \\ y_i & 1 \end{pmatrix} \succeq 0 & \text{if } a_i = 0 \end{cases}$$

Then, obviously, $y_i^2 \leq y_{i-1}x^{a_i}$, and if $y_1,\dots,y_{p-1}$ are submitted to no other constraints, we have:

$$y_p \leq x^{\sum_{i=1}^{i=p} a^i/2^{p+1-i}} = x^{k/2^p}.$$

Hence the result, by setting $y_p = y$. □

What about values of α greater than one? A simple solution is also given in the following.

Proposition 6 *Let x and y be two real positive numbers. It is possible, using SDP constraints, to bound x and y by the relation*

$$y \geq \frac{1}{x^\beta},$$

where $\beta = k/2^p$, with $p \in \mathbb{N}$ and $k \in \{0,\dots,2^p-1\}$.

Obviously, that solves our problem for $\alpha \in (1;2)$.

PROOF. Let z be an intermediate variable. Using proposition 5, one can set $z \leq x^\beta$. Also one can write

$$\begin{pmatrix} y & 1 \\ 1 & z \end{pmatrix} \succeq 0$$

which leads to $yz \geq 1$. Then x and y are bounded by the unique relation: $yx^\beta \geq 1$, hence the result. □

Proposition 7 *Let x and y be two real positive numbers. It is possible, using SDP constraints, to bound x and y by the relation*

$$y \geq \frac{1}{x^{1/\beta}},$$

where $\beta = k/2^p$, *with* $p \in \mathbb{N}$ *and* $k \in \{0,\ldots,2^p-1\}$.

That proposition covers the cases $\alpha \in (2;+\infty)$.
PROOF. Similarly, we obtain $xy^{\beta} \geq 1$. □

3.2 Proportional Fairness

In this section, we focus on the case $\alpha = 1$. The result relies on the following proposition.

Proposition 8 *Let y, and* $x_1,\ldots,x_n$ *be real positive numbers. Then using SDP constraints, it is possible to bound these numbers by the relation*

$$y^{2^{\lceil \log_2(n) \rceil}} \leq \prod_{i=1}^{i=n} x_i.$$

Then maximizing y leads immediately to the solution of our telecommunication problems with $\alpha = 1$.
PROOF. Let p be the smallest integer such that $2^p \geq n$. We construct a family of real positive variables $y_{i2^k+1,(i+1)2^k}$ with $1 \leq k \leq p$, $i \in \{0,\ldots,2^{p-k}-1\}$ $l \geq 0$, satisfying the following constraints:

$$\begin{pmatrix} y_{2i2^{k-1}+1,(2i+1)2^{k-1}} & y_{i2^k+1,(i+1)2^k} \\ y_{i2^k+1,(i+1)2^k} & y_{(2i+1)2^{k-1}+1,(2i+2)2^{k-1}} \end{pmatrix} \succeq 0,$$

where we note $y_{j,j} = x_j$ for $j \in \{1,\ldots,n\}$, and $y_{j,j} = 1$ for $j \in \{n+1,\ldots,2^p\}$. □

4 Other Usages of SDP in the Telecommunication Context

Here we mention some other fields of interest of semidefinite programming in the telecommunication context.

Clustering problems aim at finding partitions of nodes that are not two distants one from another. Let V be a set of n vertices, that we try to partition into subsets of exactly m nodes (say $k = n/p$ is an integer). Then we try to find matrices having a shape of the form

$$Y = T \begin{pmatrix} J_m & & 0 \\ & \ddots & \\ 0 & & J_m \end{pmatrix} T^t,$$

where T is a transposition matrix. Such a matrix emphasizes the partition of V into clusters, and allows to derive easily (with linear functions) the sum of the weights of the inter-domain edges for instance. In [4] the authors consider the following relaxed constraints on Y:

$$Y \succeq 0,\ diag(Y) = e,\ Ye = me,$$

and show that it provides a fair approximation for telecommunication problems, associated with reasonnable computational time.

The Max-Clique problem consists in finding a clique with maximum weight. Given a set of vertives V associated with weights $w_v,\ v \in V$, and a set of edges E, one tries to solve

$$\begin{cases} Min & \sum_{u \in V} w_u x_u & \\ s.c. & x_u + y_u \geq 1 & (u,v) \in E \\ & x_u \in \{0,1\} & u \in V \end{cases}$$

An alternative integer programming formulation of this problem is:

$$\begin{cases} Min & \sum_{u\in V} w_u(1+y_0y_u) & \\ s.c. & (y_0-y_u)(y_0-y_v)=0 & (u,v)\in E \\ & y_u\in\{-1,1\} & u\in V \\ & y_0\in\{-1,1\} & \end{cases}$$

and the SDP relaxation of this problems turns to select $y_u \in \mathbb{R}^n$, $u \in V$ and $y_0 \in \mathbb{R}^n$, and solves:

$$\begin{cases} Min & \sum_{u\in V} w_u(1+y_0\cdot y_u) & \\ s.c. & (y_0-y_u)\cdot(y_0-y_v)=0 & (u,v)\in E \\ & y_u^2=1 & u\in V \\ & y_0^2=1 & \end{cases}$$

This relaxation, also directly connected to the Lovasz theta function [3, 5], shows a basic idea of SDP programming: vectors can be used to relax binary choices with an excellent approximation.

Acknowledgements

I am particularly grateful to Alexandre Laugier for having introduced me to the field of SDP, and his help for the preparation of this draft.

References

[1] G. Anderson, L. Engebretsen, and J. Hastad. A new way to use semidefinite programming with applications to linear equations mod p. *Academic Press*, 2001. Submitted.

[2] S. Boyd, L. El Ghaoui, E. Feeron, and V. Balakrishnan. *Linear matrix inequalities in system and control theory*. Springer, 1994.

[3] J. Kleinberg and M. Goemans. The lovasz theta function and a semi-definite programming relaxation of vertex cover. *SIAM J. Discrete Math*, 11, 1998.

[4] A. Lisser and F. Rendl. Telecommunication clustering using linear and semidefinite programming. *Mathematical Programming*, 2001. To appear.

[5] L. Lovász. On the shannon capacity of a graph. *IEEE transaction on Information Theory*, 25, 1979.

[6] J. Mo and J. Walrand. Fair end-toend window-based congestion control. In *SPIE'98, International symposium on voice, video and data communications*, 1998.

[7] A. Nemirovski. What can be expressed via conic quadratic and semidefinite programming. *Lecture notes*.

[8] M.V. Ramana. An exact duality theory for semi-definite programming and its complexity implications. *Math Programming*, 1996.